

Amendments to the Claims:

This listing of claims will replace all prior versions, and listings, of claims in the application:

1. (Original) A machine-implemented method, comprising:
receiving an identifier as a parameter of a first method of an application programming interface;
in response to receiving the identifier as a parameter of the first method, invoking a utility that exports, as one or more transportable tablespaces, one or more database objects that are associated with the identifier in a database;
receiving the identifier as a parameter of a second method of the application programming interface; and
in response to receiving the identifier as a parameter of the second method, invoking a utility that imports, into the database, database objects that are contained in the one or more transportable tablespaces, and associates, with the identifier, the database objects that are contained in the one or more transportable tablespaces.
2. (Original) The method of claim 1, further comprising:
receiving a particular password as a parameter of the first method of the application programming interface; and
determining whether the particular password matches a password that is associated with the identifier;

wherein the invoking of the utility that exports one or more database objects as one or more transportable tablespaces is performed in response to a determination that the particular password matches the password that is associated with the identifier.

3. (Original) The method of claim 1, further comprising:
receiving, as a parameter of the first method of the application programming interface, an indication of a type of exporting of data to be performed; and
determining whether the indication indicates that transportable tablespaces are to be used to export data;
wherein the invoking of the utility that exports one or more database objects as one or more transportable tablespaces is performed in response to a determination that the indication indicates that transportable tablespaces are to be used to export data.

4. (Original) The method of claim 1, further comprising:
determining whether the one or more database objects that are associated with the identifier in the database contain a reference to a database object that is not associated with the identifier in the database;
wherein the invoking of the utility that exports one or more database objects as one or more transportable tablespaces is performed in response to a determination that the one or more database objects that are associated with the identifier in the database do not contain any reference to any database object that is not associated with the identifier in the database.

5. (Original) The method of claim 1, further comprising:
storing, in an export metafile, a location of a transportable tablespace metadata dump file
that is generated by the utility that exports, as one or more transportable
tablespaces, the one or more database objects that are associated with the
identifier in the database.
6. (Original) The method of claim 1, further comprising:
storing, in an export metafile, a list of one or more transportable tablespace names that
were exported by the utility that exports.
7. (Original) The method of claim 1, further comprising:
storing, in an export metafile, an indication that the one or more database objects that are
associated with the identifier in the database were exported using a type of export
that exports database objects as transportable tablespaces.
8. (Original) The method of claim 1, further comprising:
receiving the identifier as a parameter of a third method of the application programming
interface;
in response to receiving the identifier as a parameter of the third method, reading, from an
export metafile that is associated with the identifier, a location of a transportable
tablespace metadata dump file, location of associated data files, and names of one
or more transportable tablespaces that are associated with the identifier in the
database.
9. (Original) A machine-readable medium, comprising:

instructions for causing one or more processors to receive an identifier as a parameter of a first method of an application programming interface;

instructions for causing one or more processors to invoke, in response to receiving the identifier as a parameter of the first method, a utility that exports, as one or more transportable tablespaces, one or more database objects that are associated with the identifier in a database;

instructions for causing one or more processors to receive the identifier as a parameter of a second method of the application programming interface; and

instructions for causing one or more processors to invoke, in response to receiving the identifier as a parameter of the second method, a utility that imports, into the database, database objects that are contained in the one or more transportable tablespaces, and associates, with the identifier, the database objects that are contained in the one or more transportable tablespaces.

10. (Original) The machine-readable medium of claim 9, further comprising:

instructions for causing one or more processors to receive a particular password as a parameter of the first method of the application programming interface; and

instructions for causing one or more processors to determine whether the particular password matches a password that is associated with the identifier;

wherein the instructions for causing one or more processors to invoke the utility that exports one or more database objects as one or more transportable tablespaces are performed in response to a determination that the particular password matches the password that is associated with the identifier.

11. (Original) The machine-readable medium of claim 9, further comprising:
instructions for causing one or more processors to receive, as a parameter of the first
method of the application programming interface, an indication of a type of
exporting of data to be performed; and
instructions for causing one or more processors to determine whether the indication
indicates that transportable tablespaces are to be used to export data;
wherein the instructions for causing one or more processors to invoke the utility that
exports one or more database objects as one or more transportable tablespaces are
performed in response to a determination that the indication indicates that
transportable tablespaces are to be used to export data.
12. (Original) The machine-readable medium of claim 9, further comprising:
instructions for causing one or more processors to determine whether the one or more
database objects that are associated with the identifier in the database contain a
reference to a database object that is not associated with the identifier in the
database;
wherein the instructions for causing one or more processors to invoke the utility that
exports one or more database objects as one or more transportable tablespaces are
performed in response to a determination that the one or more database objects
that are associated with the identifier in the database do not contain any reference
to any database object that is not associated with the identifier in the database.

13. (Original) The machine-readable medium of claim 9, further comprising:

instructions for causing one or more processors to store, in an export metafile, a location

of a transportable tablespace metadata dump file that is generated by the utility

that exports, as one or more transportable tablespaces, the one or more database

objects that are associated with the identifier in the database.
14. (Original) The machine-readable medium of claim 9, further comprising:

instructions for causing one or more processors to store, in an export metafile, a list of

one or more transportable tablespace names that were exported by the utility that

exports.
15. (Original) The machine-readable medium of claim 9, further comprising:

instructions for causing one or more processors to store, in an export metafile, an

indication that the one or more database objects that are associated with the

identifier in the database were exported using a type of export that exports

database objects as transportable tablespaces.
16. (Original) The machine-readable medium of claim 9, further comprising:

instructions for causing one or more processors to receive the identifier as a parameter of

a third method of the application programming interface; and

instructions for causing one or more processors to read, in response to receiving the

identifier as a parameter of the third method, from an export metafile that is

associated with the identifier, a location of a transportable tablespace metadata

dump file, location of associated data files, and names of one or more

transportable tablespaces that are associated with the identifier in the database.

17. (Original) An apparatus, comprising:
- a mechanism for receiving an identifier as a parameter of a first method of an application programming interface;
 - a mechanism for invoking, in response to receiving the identifier as a parameter of the first method, a utility that exports, as one or more transportable tablespaces, one or more database objects that are associated with the identifier in a database;
 - a mechanism for receiving the identifier as a parameter of a second method of the application programming interface; and
 - a mechanism for invoking, in response to receiving the identifier as a parameter of the second method, a utility that imports, into the database, database objects that are contained in the one or more transportable tablespaces, and associates, with the identifier, the database objects that are contained in the one or more transportable tablespaces.
18. (Original) The apparatus of claim 17, further comprising:
- a mechanism for receiving a particular password as a parameter of the first method of the application programming interface; and
 - a mechanism for determining whether the particular password matches a password that is associated with the identifier;
- wherein the mechanism for invoking the utility that exports one or more database objects as one or more transportable tablespaces operates in response to a determination that the particular password matches the password that is associated with the identifier.

19. (Original) The apparatus of claim 17, further comprising:
- a mechanism for receiving, as a parameter of the first method of the application programming interface, an indication of a type of exporting of data to be performed; and
- a mechanism for determining whether the indication indicates that transportable tablespaces are to be used to export data;
- wherein the mechanism for invoking the utility that exports one or more database objects as one or more transportable tablespaces operates in response to a determination that the indication indicates that transportable tablespaces are to be used to export data.
20. (Original) The apparatus of claim 17, further comprising:
- a mechanism for determining whether the one or more database objects that are associated with the identifier in the database contain a reference to a database object that is not associated with the identifier in the database;
- wherein the mechanism for invoking the utility that exports one or more database objects as one or more transportable tablespaces operates in response to a determination that the one or more database objects that are associated with the identifier in the database do not contain any reference to any database object that is not associated with the identifier in the database.
21. (Original) The apparatus of claim 17, further comprising:
- a mechanism for storing, in an export metafile, a location of a transportable tablespace metadata dump file that is generated by the utility that exports, as one or more

transportable tablespaces, the one or more database objects that are associated with the identifier in the database.

22. (Original) The apparatus of claim 17, further comprising:

a mechanism for storing, in an export metafile, a list of one or more transportable tablespace names that were exported by the utility that exports.

23. (Original) The apparatus of claim 17, further comprising:

a mechanism for storing, in an export metafile, an indication that the one or more database objects that are associated with the identifier in the database were exported using a type of export that exports database objects as transportable tablespaces.

24. (Original) The apparatus of claim 17, further comprising:

a mechanism for receiving the identifier as a parameter of a third method of the application programming interface;

a mechanism for reading, in response to receiving the identifier as a parameter of the third method, from an export metafile that is associated with the identifier, a location of a transportable tablespace metadata dump file, location of associated data files, and names of one or more transportable tablespaces that are associated with the identifier in the database.

25. (New) A system comprising:

a base program;

a database server that is embedded within said base program;

a database; and

a plurality of add-on programs;

wherein said database server causes data to be stored, retrieved, and modified in said database;

wherein each add-on program of said plurality of add-on programs causes data to be stored, retrieved, and modified in said database;

wherein said database comprises (a) a base program section and (b) a plurality of add-on program sections;

wherein said base program section stores database objects that are owned by said base program;

wherein each add-on program section of said plurality of add-on program sections stores database objects that are owned by a separate add-on program of said plurality of add-on programs;

wherein, for each particular section of said base program section and said plurality of add-on program sections, (a) said particular section is associated a database username that is different from any database username that is associated with any other section, (b) database objects in said particular section are associated with a database username that is associated with said particular section, and (c) said particular section is associated with a password that is separate from any password that is associated with any other section;

wherein each particular add-on program of said plurality of add-on programs is prevented from accessing database objects within any section of said base program section and said plurality of add-on program sections unless said particular program

supplies a password that is associated with a section, of said base program section and said plurality of add-on program sections, that contains database objects that said particular program is attempting to access;

wherein said base program implements an application programming interface (API);

wherein said API implements methods that are invocable by each add-on program of said plurality of add-on programs;

wherein said API provides, to each add-on program of said plurality of add-on programs, a uniform set of functions that are accessible through a uniform interface;

wherein an uninstallation of said database server requires migration of database objects that are stored in said database;

wherein, by invoking methods of said API, each separate add-on program of said plurality of add-on programs migrates database objects that said separate add-on program owns;

wherein, as part of an uninstall mechanism that uninstalls an old version of a first add-on program of said add-on programs, said first add-on program invokes a first method of said API;

wherein, in invoking said first method of said API, said first add-on program specifies, as a parameter of said first method, a first database username that is associated with a first add-on program section of said plurality of add-on program sections;

wherein, in response to being invoked with said first database username as a parameter, said first method exports, from said database, as transportable tablespaces, database objects that are contained in said first add-on program section;

wherein, as part of an install mechanism that installs a new version of said first add-on program, said first add-on program invokes a second method of said API;

wherein said second method differs from said first method;

wherein, in invoking said second method of said API, said first add-on program specifies, as a parameter of said second method, said first database username that is associated with said first add-on program section of said plurality of add-on program sections;

wherein, in response to being invoked with said first database username as a parameter, said second method imports, into said first add-on program section of said database, particular database objects that are contained in said transportable tablespaces;

wherein, as a part of said install mechanism, after said particular database objects have been imported into said database from said transportable tablespaces, said first add-on program modifies said particular database objects;

wherein, before said first add-on program modifies said particular database objects, said particular database objects conform to a first schema;

wherein, after said first add-on program modifies said particular database objects, said particular database objects conform to a second schema that differs from said first schema.